

Curs 10

Liste de acces



Cuprins

- Ce este un ACL?
- Funcționarea ACL-urilor
- Tipuri de liste de acces
- Exemple de configurare



Ce este un Access List?

- Un **set de condiții** specificate de către administrator pentru **identificarea** unor anumite tipuri de trafic
- Traficul identificat poate fi
 - Filtrat
 - Alterat
 - Controlat
 - Asociat cu alte acțiuni
- În funcție de acțiunea dorită, traficul trebuie identificat după anumite criterii

Utilități ale ACL-urilor

- Filtrarea și monitorizarea traficului
 - Cea mai des folosită aplicație a ACL-urilor
 - Remember **iptables -t filter**
 - Permitearea sau respingerea traficului
 - Inspecția mai avansată a traficului identificat

Utilități ale ACL-urilor

- Marcarea și alterarea traficului
 - Remember **iptables -t mangle** and **-t nat**
 - QoS
 - Pasul 1: traffic tagging
 - Pasul 2: traffic policing și traffic shaping
 - NAT
 - Criptare

Utilități ale ACL-urilor

- Asocierea cu accesul la alte servicii
 - Accesul la terminale virtuale (ssh/telnet/http)
 - Controlul actualizărilor protocoalelor de rutare
 - Policy based routing (vom vedea în curs 10)

Criteria de identificare a traficului

- Adresă IP
 - Sursă
 - Destinație
- Protocol
 - IPv4, IPv6, IPX, AppleTalk
 - TCP, UDP
 - ICMP
- Port sau tip
 - Port sursă sau destinație la TCP sau UDP
 - Tip de mesaj ICMP

Liste de acces pentru filtrare



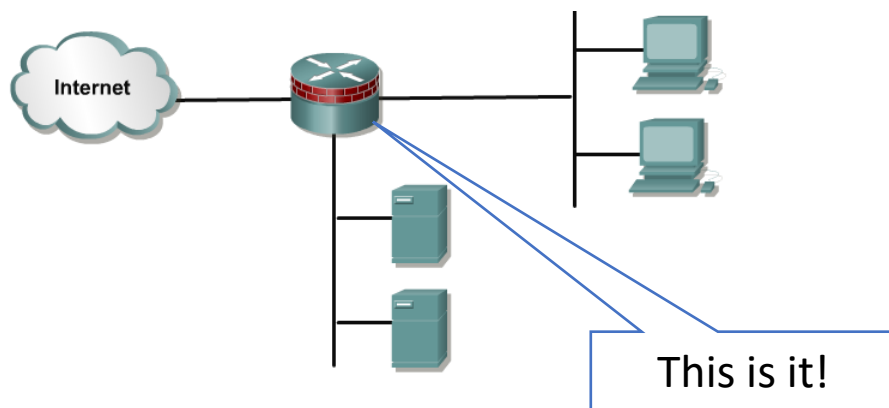
Dezavantaje?

- Timp de latență mai mare
- Încărcare suplimentară a echipamentului

| | Router dedicat | Firewall dedicat |
|---------------------------|--|--|
| Principala funcție | Rutare | Filtrare |
| Alte funcții | Permite implementarea funcțiilor de filtrare | Poate ruta, dar suportă mult mai puține facilități |
| Criptare | Nu oferă implicit | Criptare HW la rate foarte mari |
| Luare de decizii | Protocoale de nivel 3 și 4 | Protocoale de nivel 3-7 |
| SSH | - | Server SSH integrat |

Dar ce este un Firewall?

- Un firewall constă în una sau mai multe mașini care au ca scop prevenirea accesului neautorizat la o rețea.



- Acestea controlează accesul la servicii atât **din** cât și **în** rețeaua internă
- ACL-urile sunt folosite pentru a crea firewall-uri între rețeaua internă și cea externă
- **Demilitarized Zone** (DMZ) conține servicii disponibile din Internet
- Rutele firewall trebuie plasate între rețeaua internă și lumea exterioară

Definiția unui ACL

- O listă de acces conține intrări/reguli pentru controlul accesului
- Fiecare regulă
 - Identifică diferite tipuri de trafic pe baza unor criterii
 - Specifică acțiunea care trebuie luată în cazul în care criteriul a fost îndeplinit (există match)
 - Permite traficul : *permit*
 - Oprește traficul : *deny*

Parcurgerea unui ACL

- Regulile sunt testate secvențial, linie cu linie, de sus în jos, până se găsește o regulă care să facă match, sau până la sfârșitul listei
 - La match, se aplică acțiunea, și restul ACL-ului nu se mai verifică

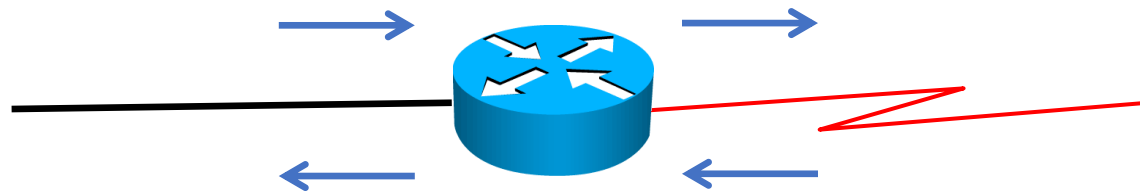
!Dacă nu se găsește niciun match, se ajunge la finalul fiecărui ACL , unde există un implicit *deny any*

Aplicarea unui ACL

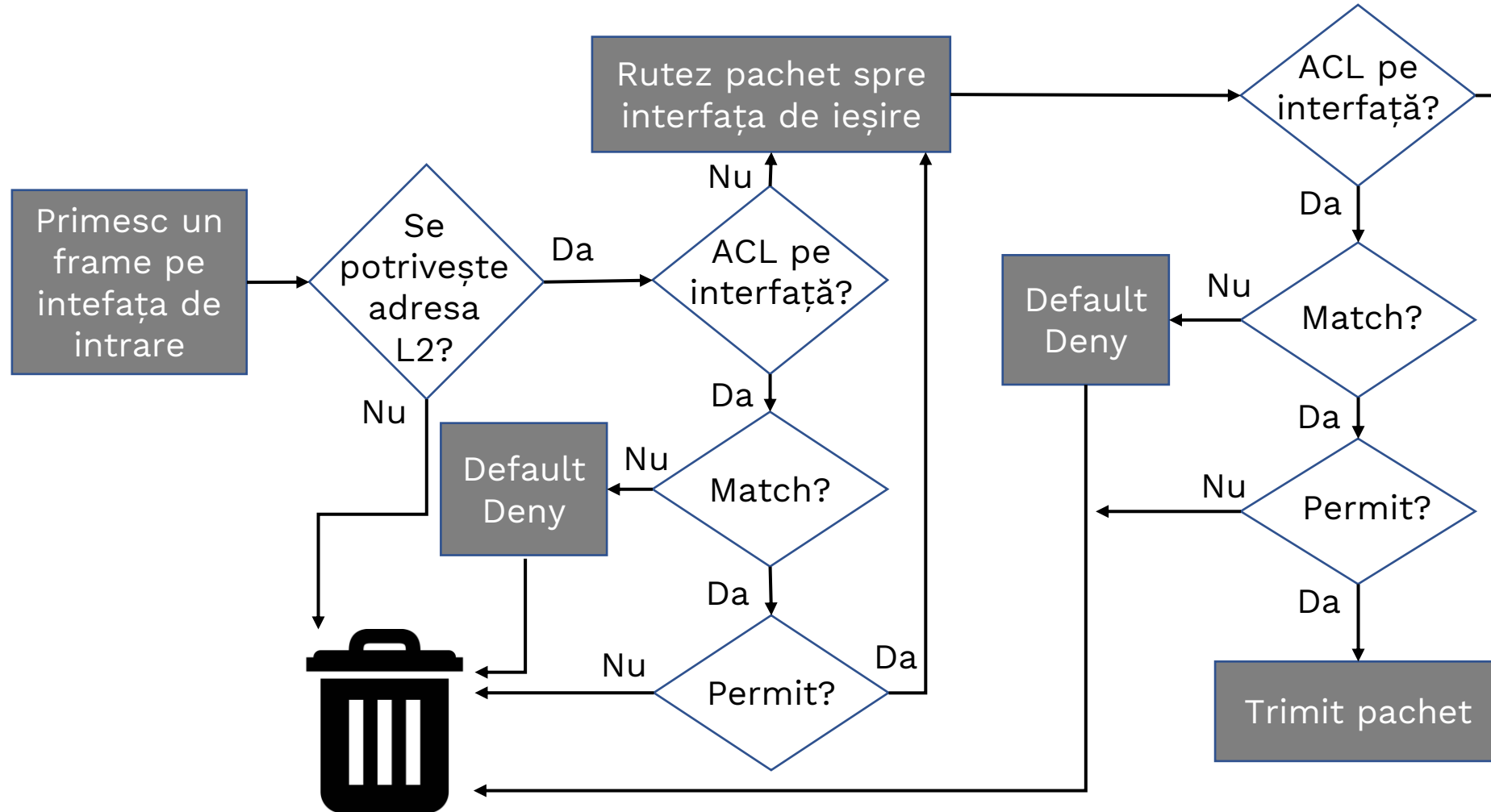
- **ACL**-urile de filtrare se pot aplica
 - Pentru fiecare protocol rutat de layer 3 (IP, IPv6 etc.)
 - Pentru fiecare interfață
 - Pentru fiecare direcție
 - Inbound, pentru traficul ce intră
 - Outbound, pentru traficul ce iese

Exercițiu: Aplicarea unui ACL

- Un ruter cu 2 interfețe rulează dual stack (IPv4, IPv6)
- Care este nr. maxim de ACL-uri de filtrare ce pot fi aplicate?
 - R: 2 (interfețe) x 2 (protocoale rutate) x 2 (in și out)



Funcționarea ACL-urilor



Tipuri de liste de acces

- Liste de acces standard
- Liste de acces extinse
- Liste de acces cu nume
 - Standard
 - Extinse
- Reflexive ACLs
- Time-based ACLs

Tipuri de liste de acces

- Liste de acces standard
- Liste de acces extinse
- Liste de acces cu nume
 - Standard
 - Extinse
- Reflexive ACLs
- Time-based ACLs

- Identificate printr-un număr între 1 și 99, sau 1300-1999 în IOS-urile mai recente
- Acceptă sau respinge o întreagă suită de protocoale
- Verifică doar **sursa pachetului**
- Trebuie plasate în rețea cât mai aproape de **destinație**.

Tipuri de liste de acces

- Liste de acces standard
- Liste de acces extinse
- Liste de acces cu nume
 - Standard
 - Extinse
- Reflexive ACLs
- Time-based ACLs

- Identificate printr-un număr între 100 și 199, sau 2000-2699 pentru IOS-urile recente
- Pot accepta sau respinge un protocol specific
- Verifică sursa pachetului, destinația, protocolul sau chiar portul
- Trebuie plasat în rețea cât mai aproape de **sursă**.

Tipuri de liste de acces

- Liste de acces standard
- Liste de acces extinse
- Liste de acces cu nume
 - Standard
 - Extinse
- Reflexive ACLs
- Time-based ACLs

- Identificate printr-un nume configurat de administrator
- Pot fi **standard sau extinse**
- Oferă flexibilitate mai mare decât listele clasice standard sau extinse
- Recomandate să fie folosite față de cele clasice

Wildcard mask

- O mască ce se suprapune peste o adresă IP
- Identifică partea comună a unor adrese IP
- Reprezintă un șir de 32 de biți de 1 și 0
 - **Bitul 0** – face match
 - **Bitul 1** – ignoră valoarea bitului din IP
- Poate fi privită ca și inversul măștii de rețea, însă poate fi folosită și pentru a identifica altfel

Wildcard mask

- Se pot folosi 2 cuvinte cheie în ACL-uri:
 - **any** – înseamnă adresa IP 0.0.0.0 și WM 255.255.255.255, toate IP-urile vor face match
 - **host** – testează egalitatea cu o adresă de host, echivalent cu WM 0.0.0.0

Wildcard mask - exemplu

- În acest exemplu, ruterul va verifica doar primii 16 biți din adresele IP și îi va compara cu cei din adresa IP. Această declarație va permite traficul având ca sursă 172.16.*.*
 - Biții de 0 – fac match
 - Biții de 1 – sunt ignorați

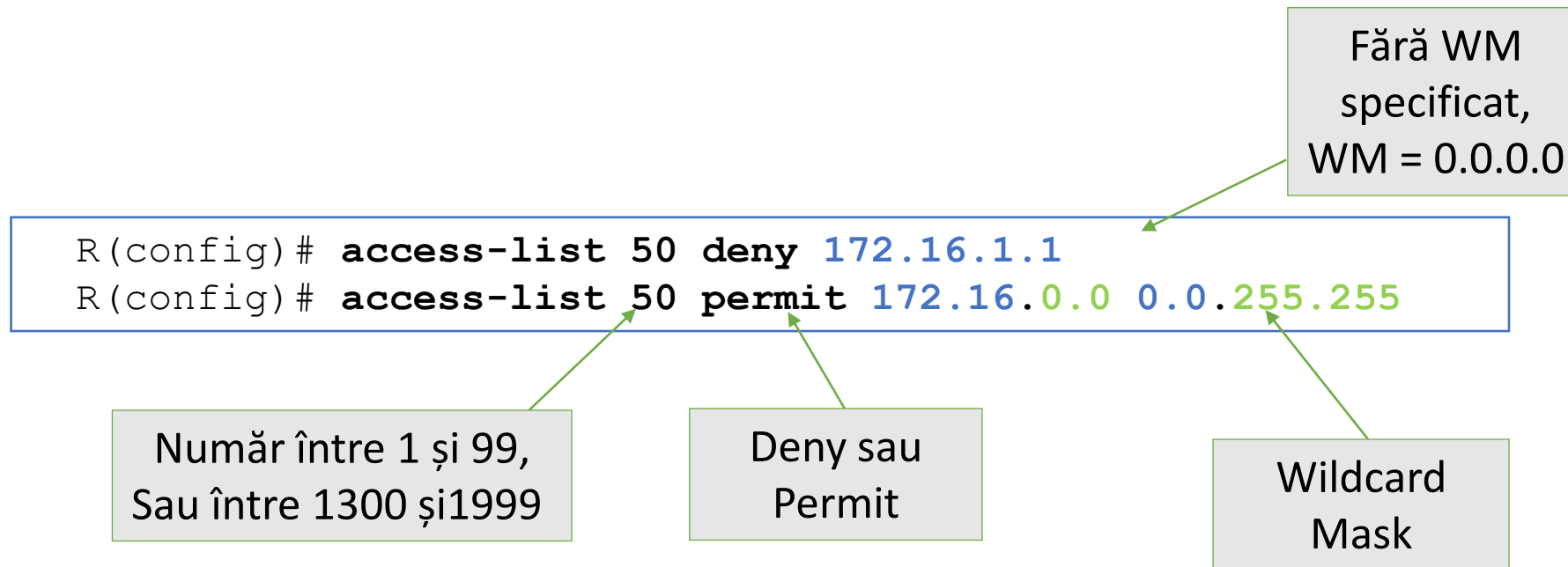
172.16.0.0 10101100.00010000.00000000.00000000
 0.0.255.255 00000000.00000000.11111111.11111111

ACL-uri clasice

- Standarde sau Extinse
 - Tipul este dat de numărul (ID-ul) listei
- Grupate în funcție de numărul (ID) comun
- Adăugate linie cu linie, dar întotdeauna la sfârșit
- Nu se poate șterge o singură linie din ACL

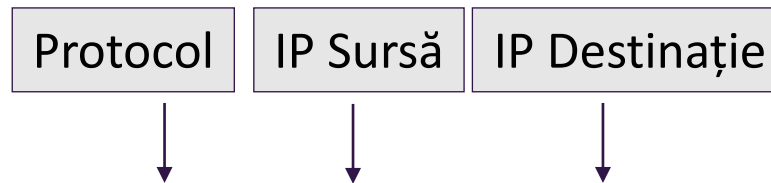
ACL-uri clasice standard

- Filtrează pachetele doar în funcție de **sursă**
- Numărul asociat unui astfel de ACL trebuie să fie între 1 și 99, sau, în versiunile mai recente de IOS, între 1300 și 1999



ACL-uri clasice extinse

- Filtrează pachetele în funcție și de **sursă** și de **destinație**. De asemenea, pot filtra pachete și în funcție de **protocol** și de **port**
- Numărul asociat unui astfel de ACL trebuie să fie între 100 și 199; în versiunile mai recente de IOS se pot folosi și numere între 2000 și 2699

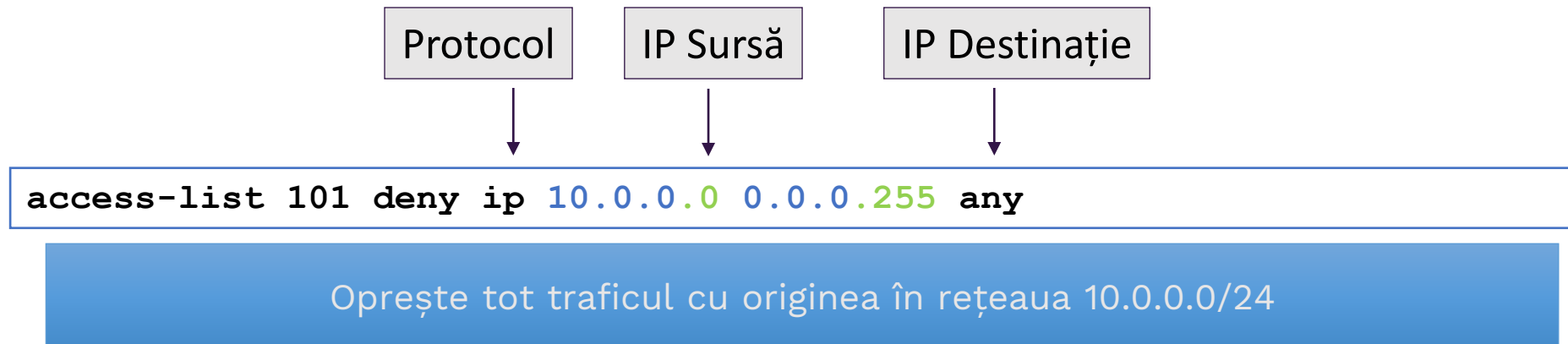


```
access-list 101 permit ip host 10.0.0.1 any
```

Permite în mod explicit tot traficul IP de la acest host către oricare altă destinație

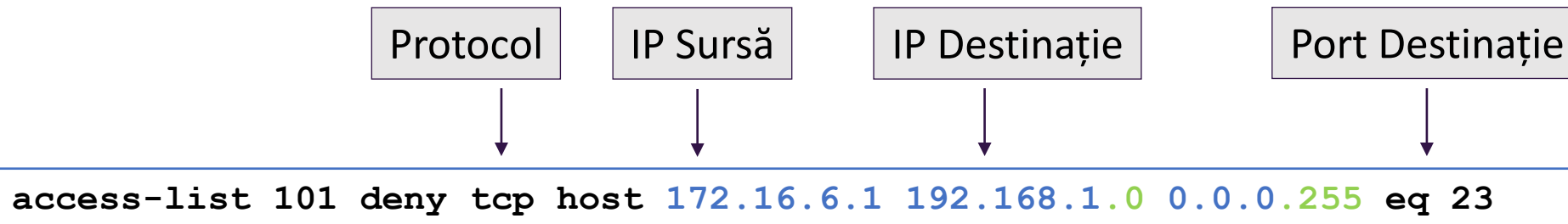
ACL-uri clasice extinse

- Filtrează pachetele în funcție și de **sursă** și de **destinație**. De asemenea, pot filtra pachete și în funcție de **protocol** și de **port**
- Numărul asociat unui astfel de ACL trebuie să fie între 100 și 199; în versiunile mai recente de IOS se pot folosi și numere între 2000 și 2699



ACL-uri clasice extinse

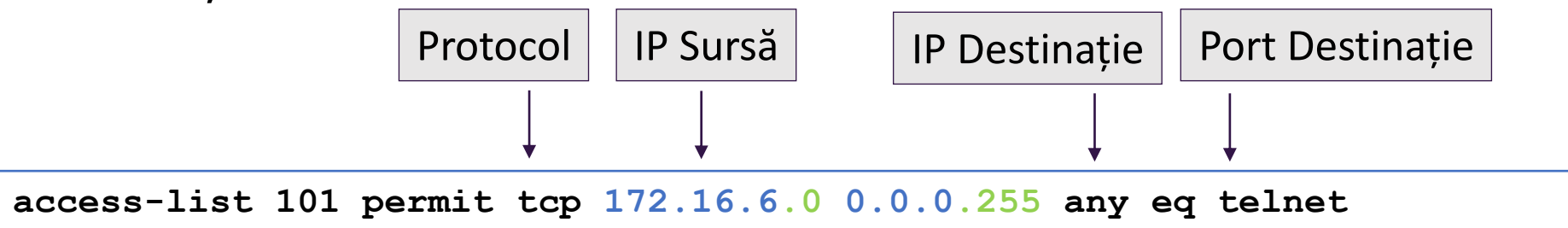
- Filtrează pachetele în funcție și de **sursă** și de **destinație**. De asemenea, pot filtra pachete și în funcție de **protocol** și de **port**
- Numărul asociat unui astfel de ACL trebuie să fie între 100 și 199; în versiunile mai recente de IOS se pot folosi și numere între 2000 și 2699



Oprește în mod explicit accesul pe portul 23(Telnet) de la host-ul 172.16.6.1 la rețeaua 192.168.1.0/24

ACL-uri clasice extinse

- Filtrează pachetele în funcție și de **sursă** și de **destinație**. De asemenea, pot filtra pachete și în funcție de **protocol** și de **port**
- Numărul asociat unui astfel de ACL trebuie să fie între 100 și 199; în versiunile mai recente de IOS se pot folosi și numere între 2000 și 2699



Permite Telnet-ul de la toate host-urile din rețeaua 172.16.6.0/24

Editarea unui ACL clasic

- Pentru a edita un ACL clasic standard sau extended:
 - Copiați ACL-ul într-un fișier text
 - Stergeți ACL-ul din fișierul de configurare al ruter-ului folosind ‘no’ și declarația ACL-ului
 - Faceți modificările necesare în fișierul text
 - Copiați pe ruter ACL-ul modificat, în global configuration mode

sau...

Named ACLs

- Nu mai sunt folosite numere pentru a diferenția ACL-uri, ci nume
 - Numele sunt mai intuitive decât numerele
 - 254 vs „DMZ_IN_FILTER”
- Este posibilă numerotarea regulilor ce sunt adăugate, pentru ca apoi să se poată face modificări fără a șterge complet lista

Named ACLs - Exemplu

```
R(config)#ip access-list extended FILTER_LAN_IN
R(config-ext-nacl)#20 permit ip any any
```

Dacă am uitat 2 reguli ce trebuiau definite înainte..

```
R(config-ext-nacl)#5 permit icmp host 10.0.0.0 any
R(config-ext-nacl)#10 deny icmp any any
```

Dacă am greșit regula de pe linia 5...

```
R config-ext-nacl)#no 5
R config-ext-nacl)#5 permit icmp host 10.0.0.1 any
```

După definire, pot aplica ACL-ul pe interfață

```
R(config)#interface fastEthernet 0/1
R(config-if)#ip access-group FILTER_LAN_IN in
```

Exemple de ACL-uri

- O listă de acces care să permită doar traficul de la stația 193.230.2.1

```
R(config)# access-list 1 permit host 193.230.2.1
                sau
R(config)# access-list 2 permit 193.230.2.1 0.0.0.0
                sau
R(config)# access-list 3 permit 193.230.2.1
```

- Soluție folosind ACL extins

```
R(config)# access-list 101 permit ip host 193.230.2.1 any
```


Exemple de ACL-uri

- Construiți și aplicați pe interfața ethernet 1 o listă de acces ce va permite doar traficul inițiat de la adresele 11.2.2.90 și 11.2.2.91.

```
R(config)# acces-list 18 permit host 11.2.2.90
R(config)# acces-list 18 permit host 11.2.2.91
                sau
R(config)# acces-list 18 permit 11.2.2.90 0.0.0.1

R(config)# interface ethernet 1
R(config-if)# ip acces-group 18 in
```

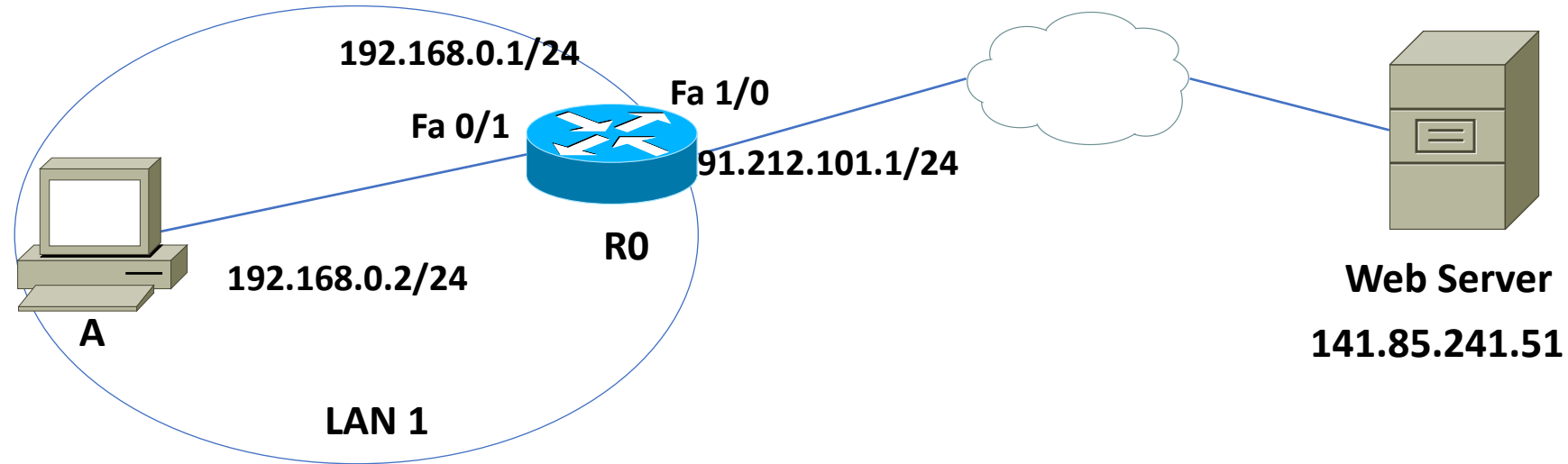
Exemple de ACL-uri

- Care este efectul următoarelor linii?

```
R(config)# interface ethernet 4
R(config-if)# ip access-group 199 out

R(config)# access-list 199 permit ip any any
R(config)# access-list 199 deny ip 106.45.0.0 0.0.255.255 any
R(config)# access-list 199 deny tcp any 44.7.12.224 0.0.0.15 eq
ftp
R(config)# access-list 199 deny udp 23.145.64.0 0.0.0.255 host
1.2.3.4 eq rip
```

Reflexive ACLs



- *Problemă* : Vrem să permitem accesul utilizatorilor din LAN 1 către Web Server, doar dacă traficul web a fost inițiat de o stație din LAN 1.

Soluție: „established”

- **established**

- opțiune pentru o regulă dintr-o listă de acces extinsă
- filtrează pachete TCP care folosesc o conexiune deja stabilită (au bitul ACK sau RST setat)

```
R0(config)#ip access-list extended ALLOW_HTTP_OUT
R0(config-ext-nacl)#10 permit tcp 192.168.0.0 0.0.0.255
any eq www
```

```
R0(config)#ip access-list extended ALLOW_HTTP_IN
R0(config-ext-nacl)#10 permit tcp host 141.85.241.51 eq
www 192.168.0.0 0.0.0.255 established
```

```
R0(config)#interface Fa1/0
R0(config-if)#ip access-group ALLOW_HTTP_OUT out
R0(config-if)#ip access-group ALLOW_HTTP_IN in
```

Dezavantaje „established”

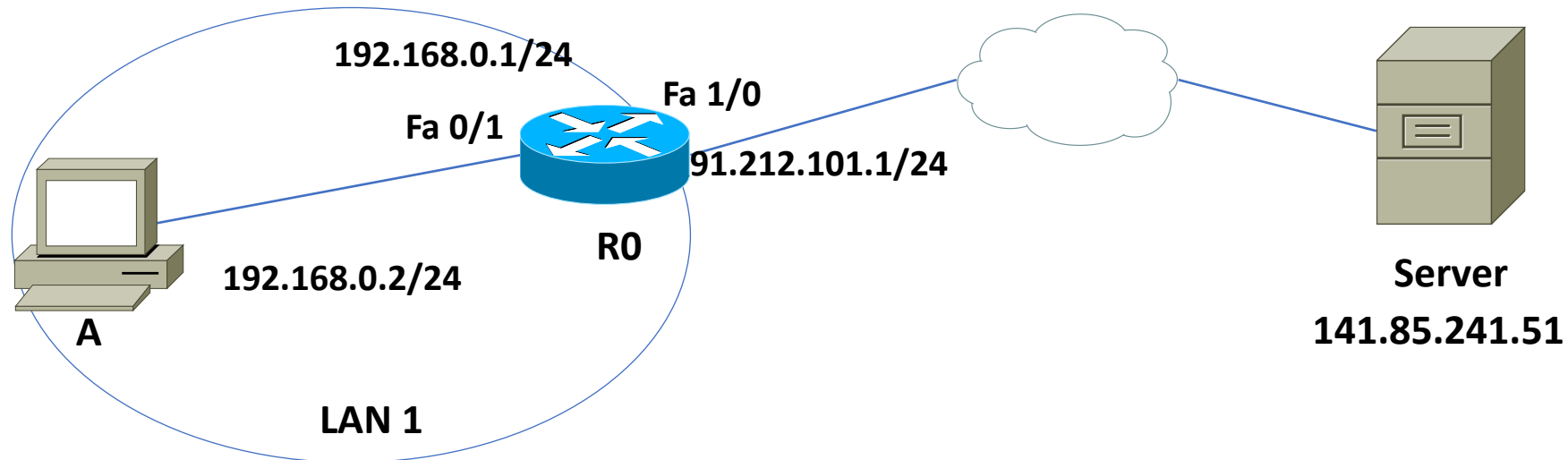
- Se verifică doar ACK și RST
- Funcționează doar pentru TCP (nu le putem folosi, spre ex. pt. a permite doar traficul ICMP care a originat în LAN1)
- Nu îl putem folosi în cazul unor aplicații care alterează dinamic portul sursă

Alternativa – Reflexive ACLs

- Filtrarea traficului pe baza informațiilor de sesiune de la nivelurile superioare nivelului 3
- Se pot defini doar prin liste de acces extinse cu nume
- Utilizate în special pentru:
 - permiterea traficului *outbound* și limitarea traficului *inbound* la sesiunile care au originea în rețeaua ruterului pe care se aplică ACL-ul reflexiv

„Reflect” și „Evaluate”

- ACL-urile reflexive – ACL-uri create dinamic pe baza unor reguli dintr-un ACL extins care au keyword-ul „*reflect*”
- *Reflect* trebuie asociat cu o regulă ce conține keyword-ul „*evaluate*”
 - *evaluate* forțează parcurgerea regulilor cu *reflect* și construirea ACL-ului dinamic care corespunde traficului ce vine ca răspuns la acestea



- *Problemă* : Vrem să permitem accesul utilizatorilor din LAN 1 către server pentru trafic HTTP și ICMP, doar dacă traficul a fost inițiat de o stație din LAN 1.

Reflexive ACLs – Exemplu

- Definirea ACL-urilor

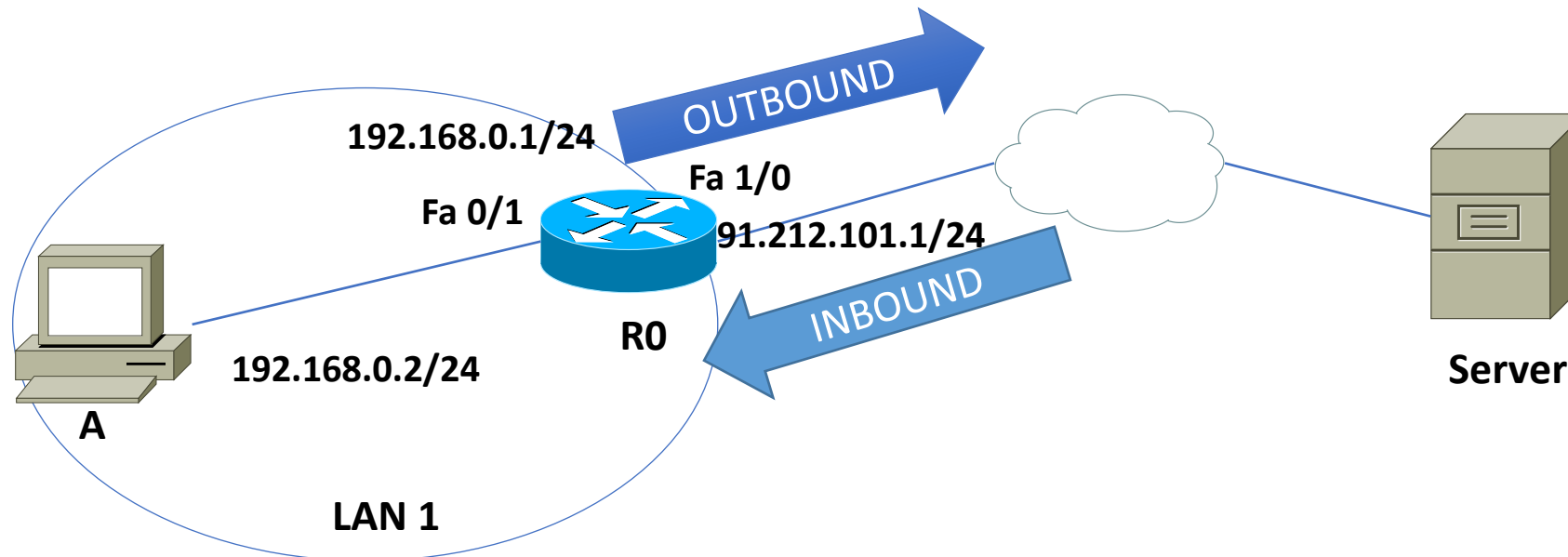
```
R0(config)#ip access-list extended OUTBOUND
R0(config-ext-nacl)#10 permit tcp 192.168.0.0 0.0.0.255 host
91.212.101.2 eq www reflect HTTPTRAFFIC
R0(config-ext-nacl)#20 permit icmp 192.168.0.0 0.0.0.255 host
91.212.101.1 reflect ICMPPTRAFFIC

R0(config)#ip access-list extended INBOUND
R0(config-ext-nacl)#20 evaluate HTTPTRAFFIC
R0(config-ext-nacl)#30 evaluate ICMPTRAFFIC
```

Reflexive ACLs - Exemplu

- Aplicarea ACL-urilor pe interfață :

```
R0(config)#interface FastEthernet1/0
R0(config-if)#ip access-group OUTBOUND out
R0(config-if)# ip access-group INBOUND in
```



Time-based ACLs

- ACL-uri care se aplică în funcție de o constrângere temporală
- Se definește un interval de timp în care ACL-ul respectiv va fi aplicat
- Atât ACL-urile clasice (numbered ACLs), cât și cele cu nume (named ACLs) acceptă definirea constrângerilor temporale

- Comenzi pentru crearea de ACL-uri time-based:

- Crearea unui interval de timp :

```
time-range time_range_name
```

- Definirea intervalului temporal :

```
periodic day(s)_of_week hh:mm to [day(s)_of_week] hh:mm
```

- sau

```
absolute start hh:mm DD Month YYYY end hh:mm DD Month YYYY
```

- Folosirea intervalului de timp într-un ACL (numbered sau named):

```
access-list <number> <extended_definitions> time-range time_range_name
```

```
ip access-list extended <name>  
<extended_definition> time-range time_range_name
```

Time-based ACLs - exemplu

- Permiteea conexiunilor de telnet doar în timpul zilelor lucrătoare:

```
R(config)#time-range work_week
R(config-time-range)#periodic Monday 9:00 to Friday 18:00

R(config)#ip access-list ext timed_acl
R(config-ext-nacl)#10 permit tcp any 192.168.1.0
0.0.0.255 eq telnet time-range work_week
R(config-ext-nacl)#interface FastEthernet1/0
R(config-if)#ip address 192.168.1.1 255.255.255.0
R(config-if)#ip access-group timed_acl out
```

- Verificarea unui time-entry:

```
R#show time-range
time-range entry: work_week (active)
    periodic Monday 9:00 to Friday 18:00
    used in: IP ACL entry
```

ACL remarks

- „Comentarii” introduse într-un ACL
 - Identificarea mai rapidă a rolului regulilor ce compun ACL-ul
- Exemplu :

```
R(config)# access-list 50 remark permit traficul spre A
R(config)# access-list 50 permit 172.16.0.0 0.0.255.255
R(config)# access-list 50 remark opresc traficul spre B
R(config)# access-list 50 deny 192.168.10.15
```

- Un comentariu este limitat la 100 de caractere

Log-uri

- Generează un mesaj ce cuprinde
 - nr. listei
 - dacă a fost acceptat/respins pachetul
 - sursa
 - nr. de pachete
- Mesajul este generat pentru primul pachet care corespunde unei reguli, iar apoi la intervale de 5 minute
- Keyword-ul **optional** *log* la finalul unei intrări într-un ACL:

```
R(config)# access-list 50 permit 172.16.0.0 0.0.255.255 log
```

Verificarea ACL-urilor

- Comenzi de **show** pentru verificarea **conținutului** și pentru **poziționarea** ACL-urilor:

| Comanda | Descriere |
|----------------------------------|---|
| <code>show ip interface</code> | Informații privind numărul de ACL-uri de intrare și ieșire |
| <code>show access-list</code> | Afișează conținutul ACL-urilor configurate pe router |
| <code>show ip access-list</code> | Afișează conținutul ACL-urilor IPv4 configurate pe router |
| <code>show running-config</code> | Afișează, printre altele, poziționarea și conținutul ACL-urilor configurate |

Sumar

Ce este un
ACL?

Tipuri de
liste de
acces

Funcționarea
ACL-urilor

Exemple de
configurare